

*Mashro3na*

# Professional Grad. Project

PiTechnologies  
Ahmed Yossef





# About PiTechnologies

- ▶ Egyptian startup company
- ▶ PiTechnologies is specialized in
  - ▶ Mobile applications development
  - ▶ Web applications development
  - ▶ Embedded systems
  - ▶ Wireless communications

# About Mashro3na

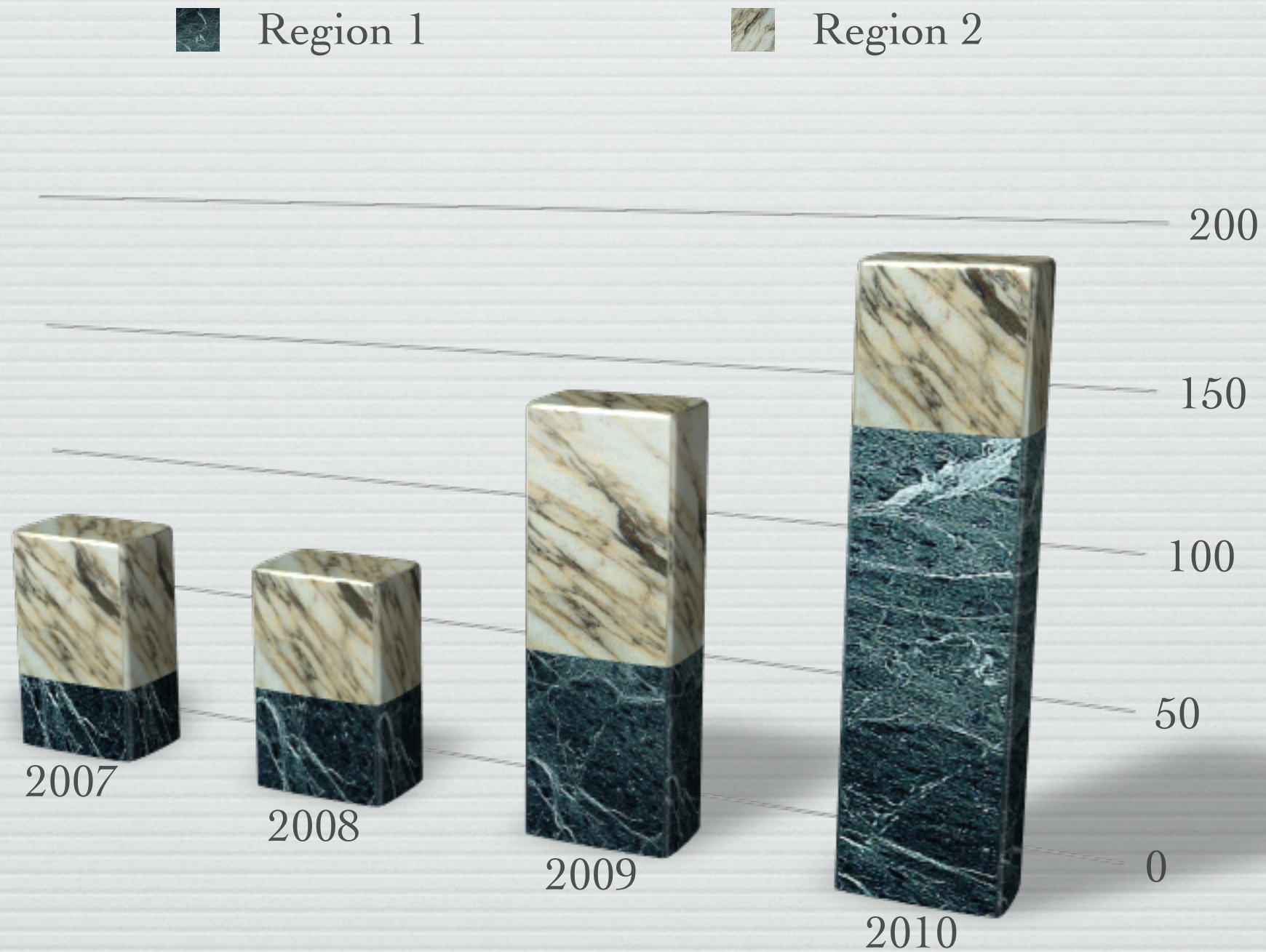
- ▶ It is an initiative of PiTechnologies
- ▶ Mashro3na aims to provide graduation projects technical and managerial support
- ▶ Mashro3na aims to enhance:
  - ▶ The graduation projects
  - ▶ The senior year student skills (technical, managerial, team working ..)



# Agenda

- What is GP ?
- Why are you working on GP ?
- GP Team
- Common mistakes
- Meetings
- Tools
  - SVN
  - Project Management





# Survey ..What is GP ?



# Main Categories

Space ship ?

Product ?

Smart Home ?

GPS + MATLAB + Java + Linux + MC + ..  
don't forget web application and mobile end :)



## Product

Low technical experience  
Waste of time connecting others  
It is not accepted as GP in most time

## Space ship

Out of scope  
Lack of support  
Ends as bike ..

Nothing new  
You are traditional for interviewer

## Smart Home

It is not a project  
You make enological things

## Assembler



Best Project

New Idea - Old Idea (added Value)

Phases

Applicable (Product)

Independent - learn technologies





Why GP ?



# Reasons

- ▶ To do something you should know why you are doing it
  - ▶ Learn new technologies
  - ▶ Apply theoretical concepts
  - ▶ Work as for real
  - ▶ Show the world something





# GP Team



# Team work

▶  $1 + 1 = 2$

Accepted

▶  $1 + 1 = 3$

Expected

▶  $1 + 1 = 0$

Actually



$$1 + 1 = 3$$

- ▶ Different tasks require different skills
- ▶ Different players .. different minds .. new solutions
- ▶ More motivation
- ▶ Work review
- ▶ Focus more on similar tasks



$$1 + 1 = 0$$

- ▶ I will work on all tasks .. me too
- ▶ i Think .. you work
- ▶ More arguments .. More problems
- ▶ You discover my errors .. i will discover yours
- ▶ I am working on all tasks .. again



# Solutions

- ▶ Changing the previous attitudes
- ▶ It is business not personal
  - ▶ I don't like your idea .. but i respect you
  - ▶ You are right your idea is better than mine
  - ▶ I don't like team decision .. i will execute it
  - ▶ I discuss only when it is good for my team  
not to prove my point of view





# Common Mistakes



# Team Homogeneity

- ▶ All A+ team
- ▶ All F- team
- ▶ All developers team
- ▶ A+ and F- team



# Team Homogeneity

Team should be miscellaneous  
Different skills  
Same goal



# Courses *Mistake*

- ▶ Divide the project requirements to a group of courses
- ▶ Attending courses till the last moment



# Courses Mistake

Define courses you need  
Define self study you need  
Restrict courses time and dealines



# Specs Mistakes

- ▶ Undefined specs .. Undefined project .. Undefined results
- ▶ Defined specs .. Undefined manners of execution



# Specs Mistakes

Idea is not enough  
You should have full specs of the system before  
starting



# Time Mistakes

- ▶ Unplanned
- ▶ Wrong plan
  - ▶ Too much time for learning
  - ▶ Too much time for working
  - ▶ No backup plan and no margins



# Time Mistakes

Time should be well divided  
Project should be divided into phases  
Delay means missing features not project failure  
Each simple task should be planned  
Learn, Design, Implement, Test, Document  
Keep margins





# Meetings



# Teamwork and Meetings

- ▶ Meeting is the most important part of the teamwork
- ▶ As during meeting:
  - ▶ Work is divided into tasks
  - ▶ Tasks are assigned
  - ▶ Completed tasks are delivered



# Agenda

- ▶ Leader should announce the agenda before the meeting
- ▶ Members are free to add any topics before the meeting
- ▶ Leader should state the agenda as check list of tasks, decisions and deliverables

# Apology

- ▶ If you will be delayed, you should report before the meeting starts.
- ▶ If you will not be able to come, you should delegate your vote and work to some one else.



# During Meeting

- ▶ No argument
- ▶ Limited discussion
- ▶ If A and B has different point of views, each one would separately explain his, then leader should terminate discussion to start voting.
- ▶ In dark situations, leader can make a decision, and you should accept it.

# After the Meeting

- ▶ During the meeting, leader should have written notes about the discussions and the decisions.
- ▶ Leader should send MoMs to the team
- ▶ Leader should make the agenda and the time of the next meeting



# Apology for deliverables

- ▶ You should report any errors or blocking breaks just when they appear.
- ▶ Always report and make your leader updated
- ▶ Don't change your task without confirmation





# Tools - Version Control



# What is VC

- ▶ Version control - Source control - Revision control
- ▶ It is a software enables you to keep history of changing in documents, code files, ..
- ▶ It can be centralized or distributed

# Why Version Control

- ▶ No conflict
- ▶ No project\_last, project\_final, project\_finaaaaal
- ▶ A change in a single file within the project, generates a new version with number, timestamp, creator and comment
- ▶ No need to merge copies and move code on flash memory



# Check Out

- ▶ It is an operation to request a copy of the project to work on
- ▶ You can check out the latest version or any older version you like

# Check in

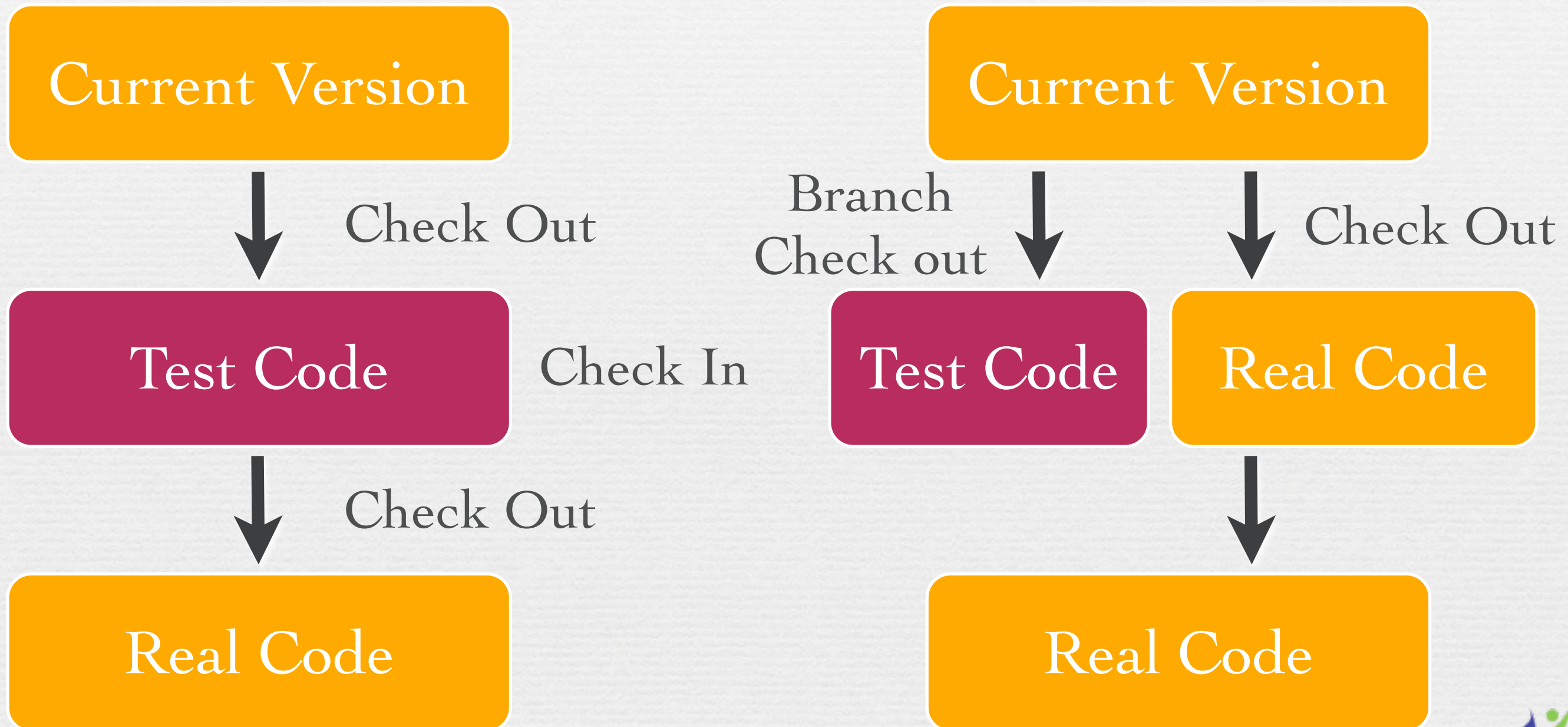
- ▶ Check in - Commit -Submit
- ▶ It is an operation done after checking out and making some modifications
- ▶ You commit your changes to save them as a new version with timestamp and comment



# Branching

- ▶ It is an operation to make a sub-project/parallel project
- ▶ You make this operation when you want to test a new feature.

# Test Scenario





# Examples

- ▶ SVN
- ▶ CVS
- ▶ <http://assembla.com/> (online)
- ▶ Git
- ▶ <https://github.com/> (online)





# Tools - Project Management



# What is PM ?

- ▶ It is the process of
  - ▶ Diving work into phases and tasks
  - ▶ Identifying execution percentage
  - ▶ Finding critical issues
  - ▶ Finding alternative solutions



# Project

Phase

Task

Task

Task

Task

Phase

Task

Task

Task

Task

Phase

Task

Task

Task

Task

# Phases

- ▶ You should divide your project into phases
- ▶ Each phase represents a percentage of total project
- ▶ Phase is a milestone, with defined date you should measure your progress



# Task

- ▶ It is the building block of the project
- ▶ Each task should have the following
  - ▶ Major phase
  - ▶ Type
  - ▶ Dead line
  - ▶ Assigned player(s)
  - ▶ Reviewer and/or supervisor

Task the basic element in the project..

If you could manage tasks .. you will manage your project .. otherwise !

# Task Types

## Planning

- To define tasks
- Involves a lot of estimation and lack of info
- Usually involves learning tasks

## Learning

- To know about something
- Undefined time
- Should has a deadline

## Execution

- To get something done.
- Defined goal
- Estimated time



# Examples

- ▶ Codendi (Web application)
- ▶ Redmine (Web application)
- ▶ Planner
- ▶ QDPM (Web application)

# Contacts



Website:

[www.pitechnologies.net](http://www.pitechnologies.net)



Fan Page:

[facebook.com/pitechnologies.page](https://facebook.com/pitechnologies.page)



Twitter:

@pitechnologiess



My mail:

ayossef@pitechnologies.net

Thanks and best of luck .. AhmedYossef